Spark 1.1.0正式发布

我们期待已久的Spark 1.1.0在美国时间的9月11日正式发布了，官方发布的声明如下：

> We are happy to announce the availability of Spark 1.1.0! Spark 1.1.0 is the second release on the API-compatible 1.X line. It is Spark's largest release ever, with contributions from 171 developers!
>
> This release brings operational and performance improvements in Spark core including a new implementation of the Spark shuffle designed for very large scale workloads. Spark 1.1 adds significant extensions to the newest Spark modules, MLlib and Spark SQL. Spark SQL introduces a JDBC server, byte code generation for fast expression evaluation, a public types API, JSON support, and other features and optimizations. MLlib introduces a new statistics libary along with several new algorithms and optimizations. Spark 1.1 also builds out Spark's Python support and adds new components to the Spark Streaming module.
>
> Visit the release notes to read about the new features, or download the release today.


微信扫一扫，加关注
即可及时了解Spark、Hadoop或者Hbase
等相关的文章
欢迎关注微信公共帐号：iteblog_hadoop

过往记忆博客(http://www.iteblog.com)
专注于Hadoop、Spark、Flume、Hbase等
技术的博客，欢迎关注。

Hadoop、Hive、Hbase、Flume等交流群：138615359和149892483

如果想及时了解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公共帐号：iteblog_hadoop

更新说明如下：

Spark 1.1.0是 1.X分支的第一个minor release。这个版本的Spark在core方面提供了很好的操作性和性能，并且在Spark最新的类库：MLlib 和 Spark SQL带来了最要的扩展。这些扩展在Spark Python 版也提供支持，并且在Spark Streaming模块上加上了新的组件。 Spark 1.1有171贡献值参加。（下面更新我先用英文发布，慢慢我翻译成中文，欢迎关注）

一、表现性和可用性的提升（Performance and Usability Improvements）

Across the board, Spark 1.1 adds features for improved stability and performance, particularly for large-scale workloads. Spark now performs disk spilling for skewed blocks during cache operations, guarding against memory overflows if a single RDD partition is large. Disk spilling during aggregations, introduced in Spark 1.0, has been ported to PySpark. This release introduces a new shuffle implementation optimized for very large scale shuffles. This "sort-based shuffle" will be become the default in the next release, and is now available to users.

For jobs with large numbers of reducers, we recommend turning this on. This release also adds several usability improvements for monitoring the performance of long running or complex jobs. Among the changes are better named accumulators that display in Spark's UI, dynamic updating of metrics for progress tasks, and reporting of input metrics for tasks that read input data.

二、Spark SQL

Spark SQL adds a number of new features and performance improvements in this release. A JDBC/ODBC server allows users to connect to SparkSQL from many different applications and provides shared access to cached tables. A new module provides support for loading JSON data directly into Spark's SchemaRDD format, including automatic schema inference. Spark SQL introduces dynamic bytecode generation in this release, a technique which significantly speeds up execution for queries that perform complex expression evaluation. This release also adds support for registering Python, Scala, and Java lambda functions as UDFs, which can then be called directly in SQL. Spark 1.1 adds a public types API to allow users to create SchemaRDD's from custom data sources. Finally, many optimizations have been added to the native Parquet support as well as throughout the engine.

三、MLlib

MLlib adds several new algorithms and optimizations in this release. 1.1 introduces a new library of statistical packages which provides exploratory analytic functions. These include stratified sampling, correlations, chi-squared tests and support for creating random datasets. This release adds utilities for feature extraction (Word2Vec and TF-IDF) and feature transformation (normalization and standard scaling). Also new are support for nonnegative matrix factorization and SVG via Lanczos. The decision tree algorithm has been added in Python and Java (https://issues.apache.org/jira/browse/SPARK-2478). A tree aggregation primitive has been added to help optimize many existing algorithms. Performance improves across the board in MLlib 1.1, with improvements of around 2-3X for many algorithms and up to 5X for large scale decision tree problems.

四、GraphX and Spark Streaming

Spark streaming adds a new data source Amazon Kinesis. For the Flume support, a new mode is support which pulls data from Flume, simplifying deployment and providing high availability. The first of a set of streaming machine learning algorithms is introduced with streaming linear regression. Finally, rate limiting has been added for streaming inputs. GraphX adds custom storage levels for vertices and edges along with improved numerical precision across the board. Finally, GraphX adds a new label propagation algorithm.

五、其他方面的提升（Other Notable Improvements）

PySpark now allows reading and writing arbitrary Hadoop InputFormats, including SequenceFiles, HBase, Cassandra, Avro, and other data sources
Stage resubmissions are now handled gracefully in the Spark UI
Spark supports tight firewall rules for all network ports
An overflow bug in GraphX has been fix that affects graphs with more than 4 billion vertices

六、更新日志

Spark 1.1.0 向后兼容 Spark 1.0.X.

一些配置选项已经变化了，这些可能会影响到目前的使用者。

1、spark.io.compression.codec属性的默认值现在设置为snappy.
之前的属性值可以直接将它设置为zf.

2、PySpark now performs external spilling during aggregations. Old behavior can be restored by setting spark.shuffle.spill to false.

3、PySpark uses a new heuristic for determining the parallelism of shuffle operations. Old behavior can be restored by setting spark.default.parallelism to the number of cores in the cluster.

七、其他方面更新

可以参照我昨天发布的文章。《Spark SQL 1.1.0和Hive的兼容说明》、《Shark迁移到Spark 1.1.0 编程指南》

本博客文章除特别声明，全部都是原创！
原创文章版权归过往记忆大数据（过往记忆）所有，未经许可不得转载。
本文链接: 【】（）